REMARKS

Reconsideration of this application is respectfully requested. Claims 4 and 9 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claims 1-22 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent Number 6,298,446 by Schreiber et al. (hereinafter "Schreiber").

Claims 4 and 9 have been amended. No claims have been canceled.

Claim Rejections - 35 USC § 112

The Examiner rejected Claims 4 and 9 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The Examiner states:

> Claims 4 and 9 both recite the limitation "the security mode" in line 2. There is insufficient antecedent basis for this limitation in the claim.

(Office Action, page 2)

Claims 4 and 9, as amended, recite "a security mode." Applicants respectfully assert that claims 4 and 9, as amended overcome the above rejection.

Claim Rejections - 35 USC § 102

The Examiner rejected claims 1-22 under 35 U.S.C. 102(e) as being anticipated by Schreiber. The Examiner asserts that one skilled in the art would reasonably interpret the claim terms "a presentation controller" and "a graphics controller" to include apparatuses such as web browser application software and any associated plug-ins.

The Examiner essentially substitutes the term 'web browser' everywhere the terms 'presentation controller' and 'graphics controller' appear in applicants' claims. Applicants respectfully traverse the Examiner's position that one skilled in the art would reasonably interpret a presentation controller to include a web browser after reviewing Schreiber. Applicants respectfully assert that claims 1-22 are not anticipated by Schreiber under 35 U.S.C. § 102(e).

Independent claim 11 states:

> 11. An apparatus, comprising:
> a presentation controller having:
> a presentation buffer;
> a command handler to process commands and addresses;
> a data handler coupled to the presentation buffer to monitor data and to pass at least a part of the data to the presentation buffer;
> a security violation detector to detect a request by a requestor to read protected data in the presentation buffer; and
> a data protector coupled to the data handler to prevent providing the protected data to the requestor.

(emphasis added)

The Examiner's position side steps clear limitations literally stated in the claims regarding the "presentation controller." The Examiner's position ignores that fact that claim 11 literally states that the presentation controller has a presentation buffer, a command handler, a data handler, a security violation detector, and a data protector. Rather, the Examiner asserts that the Web browser application software is merely associated with a system that has those components. Further, Schreiber actually teaches away from one reasonably skilled in the art considering a web browser as an implementation of either a graphics controller or a presentation controller. Schreiber discloses that the web browser is application level software that cooperates with the operating system level software to deliver its data to an actual video circuit.

Schreiber discloses that the invention is web browser software that operates differently then other anti software piracy techniques by not supplying original content in the first place.

> Thus, unlike software piracy techniques that protect an original copy of software from being illegally copied, the present invention does not provide an original copy in the first place.

(Schreiber, Col. 3, Lns. 11-14)

> Specifically, the present invention blocks copying of an image from within his web browser, when a user selects the "Save Image As . . ." command and when a user prints the contents of a web browser window.

(Schreiber, Col. 3, Lns. 17-21)

> In a preferred embodiment, the present invention uses a software web server plug-in that filters HTTP requests and sends substitute data, such as encrypted image data, for requested image data that is protected. It also uses a software web browser plug-in for displaying the substitute data and for blocking the ability to copy protected image data being displayed from the video buffer of the user's computer.

(Schreiber, Col. 3, Lns. 26-34)

Schreiber discloses that the web browser is a software application that resides on a Personal Computer.

> [A] client computer running a network browser.

(Schreiber, Abstract Paragraph)

Schreiber discloses that the web browser assembles the parts of a web page that will eventually be displayed on a display device.

> When a web browser in a client computer downloads a web page file, it parses the web page in order to display it on a video monitor. While parsing the web page, the web browser encounters the references to graphic objects, and in turn downloads the graphic objects.

(Schreiber, Col. 6, Lns. 41-46)

Schreiber discloses that the web browser software communicates with the operating system software (OS). The web browser contains additional software, i.e. a plug-in, to supply substitute functions for OS level API functions when the client computer calls for data from the video buffer to copy it to a clip board. More importantly, Schreiber discloses that its inventive web browser software is something different then the video card containing circuitry including the video buffer that actually supplies pixel data to the display device.

> However, it is apparent to those skilled in the art that in order to display a protected image within a web page, at some level within the operating system decoded pixel data has to be available. Typically, a video card displaying image data on a video monitor stores the image data within a video display buffer. As such, even if the image data is encrypted when downloaded to the client computer, within the client video buffer the data is available as raw pixel data, and at some level the encrypted data is decoded before it can be displayed.
>
> Pixel data stored within a video display buffer is susceptible to unauthorized use or copying, since an operating system typically enables a programmer to access data in the video display buffer. For example, the Windows operating system of Microsoft Corporation of Redmond, WA, provides system functions, such as the familiar BitBlt function, for accessing pixel data within the video display buffer. Moreover, such operating systems provide high level functions, such as the Print Screen function, which serve to copy data from the video display buffer to another memory buffer, such as a clipboard. Once image data has been copied to a clipboard, it can be easily saved and used for unauthorized purposes.
>
> In a preferred embodiment, the present invention prevents a user from using Windows API functions, such as BitBlt, StretchBlt, PlgBlt, GetPixel and GDI32, to copy protected image data, by including software within the user's web browser that substitutes other functions for those Windows API functions. For example, the software within the user's web browser provides a substitute BitBlt function, which is invoked instead of the standard system BitBlt function when the user issues a command to copy data from the video display buffer.

(Schreiber, Col. 17, Ln. 63 to Col. 18 Ln. 28)

> In a preferred embodiment of the present invention, software such as a Netscape plug-in or an Internet Explorer Active-X control is used to

> modify operating system function 706, by introducing additional programming logic to be used when attempting to access pixel data from protected images. Modification of operating system function 706 is preferably accomplished by providing a substitute function of the same name, which supersedes and is invoked instead of the standard system function.

(Schreiber, Col. 19, Lns. 1-9)

> At step 802 the user opens a web page in his web browser. At step 804 the client computer renders the web page including an embedded image. At step 806 the user views the web page, and at step 808 the user attempts to copy the embedded image by executing a command to copy pixel data of the image from a video buffer to a clipboard. For example, the user may execute the Print Screen or such other screen capture command.
>
> At step 810, in response, the client computer calls an operating system function, such as the Windows BitBlt function, to extract pixel data from the video buffer and copy it to the clipboard.
>
> At step 812 control logic passes to a substitute function, and a test is made as to whether or not the image data in the video buffer is protected. If so, then at step 814 processing jumps to step 818 where substitute program code replaces the requested pixel data with substitute data, and at step 820 the substitute data is returned by the operating system function. If the image data in the video buffer is not protected, then processing jumps to step 816 following step 812, and the requested pixel data is returned by the operating system function, as usual.
>
> At step 822 the data returned from the operating system function is written to the clipboard and at step 824 the user pastes the data from the clipboard into a window of another software application, or save it into his computer. Since substitute data was used to replace protected pixel data, the user is unable to copy unmodified pixel data from the protected image.

(Schreiber, Col. 19, Lns. 30-58)

Accordingly, applicants respectfully assert that one skilled in the art after reviewing applicants' specification and Schreiber would not reasonably interpret a presentation controller to include the web browser software disclosed in Schreiber. Accordingly, applicants assert that Schreiber does not disclose a presentation controller that has a presentation buffer, a command handler, a data handler, a security violation

detector, and a data protector. As such, Schreiber does not disclose each and every limitation of claim 11. As such, claim 11 is not anticipated by Schreiber under 35 U.S.C. § 102(e).

Given that claims 12-16 depend from and include the limitations of claim 11, applicants submit that claims 12-16 are not anticipated by Schreiber under 35 U.S.C. § 102(e).

Likewise, independent claim 17 states:

> 17. A system, comprising:
> a presentation circuit including:
> an input interface to receive data;
> an output port to transmit data for presentation;
> a presentation buffer coupled to the output port;
> a presentation controller coupled to the presentation buffer and to the input interface and having:
> a command handler to process commands and addresses; and
> a data handler to monitor data and to pass at least a part of the data to the presentation buffer;
> a security violation detector to detect a request by a requestor to read protected data in the presentation buffer; and
> a data protector to prevent providing the protected data to the requestor.

(emphasis added)

As discussed above, the web browser discussed in Schreiber does not disclose "a presentation circuit that includes a presentation controller coupled to the presentation buffer and to the input interface." Moreover, the Examiner's position ignores that fact that claim 17 literally states that the presentation controller is included as part of a presentation circuit and the presentation controller couples to the presentation buffer and to the input interface. The Examiner asserts that the web browser application software merely transmits a request to that presentation circuit and its associated component parts. However, the Examiner's position overlooks how web browser

software can be part of a circuit and couple to physical components within that circuit. As such, Schreiber does not disclose each and every limitation of claim 17. As such, claim 17 is not anticipated by Schreiber under 35 U.S.C. § 102(e).

Given that claims 18-22 depend from and include the limitations of claim 17, applicants submit that claims 18-22 are not anticipated by Schreiber under 35 U.S.C. § 102(e).

Independent claim 1 states:

> 1. A method, comprising:
> receiving data in a presentation buffer of a presentation controller;
> receiving a request from a requestor to read the data in the presentation buffer;
> deleting the data from the presentation buffer in response to the request; and
> not delivering the data to the requestor in response to the request.

(emphasis added)

As discussed above, Schreiber does not disclose "receiving data in a presentation buffer of a presentation controller." Applicant also traverses the Examiner's assertion that Schreiber discloses "deleting the data from the presentation buffer in response to the [read] request." The Examiner suggests that Schreiber deletes the data in the presentation buffer by purging the requested data with other data. First, Schreiber discloses that it never copies the original data to the computer. Schreiber discloses:

> The present invention enables a user to view content without being able to copy it into his computer. . . .
> Thus, unlike software piracy techniques that protect an original copy of software from being illegally copied, the present invention does not provide an original copy in the first place.
>
> (Schreiber, Col. 3, Lns. 6-14)

Thus, the original content is not stored in the presentation buffer to be deleted in response to a read request.

In contrast, Schreiber discloses that the substitute data is stored in the video buffer. Moreover, Schreiber discloses that the substitute data is given to the user upon a read request. Schreiber does not disclose any operation were the substitute data is given to the user and the data in the video buffer is deleted. Schreiber discloses:

> [T]he present invention replaces the image being saved with substitute data, so that the user in fact saves a substitute image. For example, the substitute image may be an encrypted image, which the user is unable to view. For another example, the substitute image may be a watermarked version of the original image, derived therefrom by composing watermarks over the image. For yet another example, the substitute image may be a prescribed image, possibly unrelated to the image being displayed by the web browser. Thus when the user selects the "Save Image As . . ." option, or presses the "Print Screen" button, or copies the image from another software application, the image that is saved into the local file system or copied to the clipboard is a substitute image.

(Schreiber, Col. 7, Lns. 44-57)

> Substitute data 124 preferably corresponds to an image that is visually identical or substantially similar to protected image 108. When substitute data 124 corresponds to an image that is visually identical to protected image 108, it is preferably an encrypted version of the protected image data. In a preferred embodiment of the present invention, the choice of what type of substitute data 124 to use depends on the owner's preference (e.g. whether or not to display an identical version of the protected image) and on the type of web browser 112 issuing the HTTP web page request from client computer 106.

(Schreiber, Col. 10, Lns. 12-22)

Thus, Schreiber discloses supplying substitute data upon the video buffer receiving a read request rather then deleting the data from the presentation buffer. As such, Schreiber does not disclose each and every limitation of claim 1. As such, claim 1 is not anticipated by Schreiber under 35 U.S.C. § 102(e).

Given that claims 2-5 depend from and include the limitations of claim 1, applicants submit that claims 2-5 are not anticipated by Schreiber under 35 U.S.C. § 102(e).

Likewise, independent claim 6 states:

> 6. A machine-readable medium having stored thereon instructions, which when executed by at least one processor cause said at least one processor to perform:
> receiving data in a presentation buffer of a presentation controller;
> receiving a request from a requestor to read the data in the presentation buffer;
> deleting the data from the presentation buffer in response to the request; and
> not delivering the data to the requestor in response to the request.

As discussed above, Schreiber does not disclose "receiving data in a presentation buffer of a presentation controller." Further, Schreiber does not disclose "deleting the data from the presentation buffer in response to the request." As such, Schreiber does not disclose each and every limitation of claim 6. As such, claim 6 is not anticipated by Schreiber under 35 U.S.C. § 102(e).

Given that claims 7-10 depend from and include the limitations of claim 6, applicants submit that claims 7-10 are not anticipated by Schreiber under 35 U.S.C. § 102(e).

Also, Schreiber only qualifies as prior art under 35 U.S.C. 102 (a) and 102 (e). Accordingly, applicants' reserve the right under 37 CFR 1.131 to swear behind this reference.

Conclusion

It is respectfully submitted that in view of the amendments and remarks set forth herein, the rejections and objections have been overcome. Applicant respectfully requests that a timely Notice of Allowance be issued in this case. A petition for an extension of time is submitted with this amendment. An Information Disclosure Statement is also submitted with this amendment. If there are any additional charges, please charge them to our Deposit Account No. 02-2666.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 9-20-04

Thomas S. Ferrill
Reg. No. 42,532
Tel.:(408) 720-8300

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026